

23rd International Meshing Roundtable (IMR23)

Automatic swept volume decomposition based on sweep directions extraction for hexahedral meshing

Haiyan Wu, Shuming Gao*

State Key Lab of CAD&CG, Zhejiang University, Hangzhou 310058, PR China

Abstract

Automatic and high quality hexahedral meshing of complex solid models is still a challenging task. To guarantee the quality of the generated mesh, current commercial software normally requires users to manually decompose a complex solid model into a set of simple geometry like swept volume whose high quality hexahedral mesh can be easily generated. The manual decomposition is a time-consuming process, and its effect heavily depends on the user's experience. Therefore, to automate the solid model decomposition for hexahedral meshing is of significance. However, the efficiency of the existing algorithms are still far from expected. In this paper, an automatic swept volume decomposition approach based on sweep directions extraction is presented. The approach first extracts all the potential local sweep directions (PLSDs) of a given solid model using heuristic rules, then generates a relevant face set (RFS) for each PLSD, and incrementally determines all the swept volumes including heavily interacting ones based on PLSDs. Furthermore, to make the decomposition good for high quality hexahedral meshing, the approach constructs reasonable cutting face sets (CFSs) to split the interacting swept volumes. Experimental results show the effectiveness of our approach.

© 2014 The Authors. Published by Elsevier Ltd.

Peer-review under responsibility of organizing committee of the 23rd International Meshing Roundtable (IMR23).

Keywords: volume decomposition; hexahedral mesh; mesh generation; swept volume; sweep direction.

1. Introduction

Finite element analysis (FEA) is an important tool to simulate complex physical phenomena and finite element mesh is usually required as input. High quality conformal hexahedral mesh has many desirable properties [1], therefore the generation of high quality hexahedral mesh is drawing more attention.

However, currently there is no general method which can be used to generate high quality hexahedral mesh for complicated solid model automatically. The users usually have to manually decompose the solid model into a set of simple geometry before high quality hexahedral mesh can be generated. The time consumed and the quality of the final mesh depend on the experience of the users.

* Corresponding author. Tel.: +86 571 88206681 ; fax: +86 571 88206680.

E-mail address: smgao@cad.zju.edu.cn

Since manual volume decomposition is time-consuming and needs a lot of human effort, there are many works on automatic volume decomposition for hexahedral meshing, including medial surface based methods, surface mesh based methods, feature based methods and so on.

The medial surface [2] is a representation of the original solid model with reduced dimension. Many researchers propose volume decomposition methods based on the medial surface. Price and Armstrong et al. [3,4] describe a medial surface subdivision technique and subdivide a large class of solid objects into topologically simple subregions suitable for automatic finite element meshing with hexahedral elements. The subdividing cuts are between parts of the object in geometric proximity and produce good quality meshes of hexahedral elements. Lu et al. [5,6] describe an approach that combines the volumetric decomposition suggestions and a pen-based user interface (UI) to assist in the geometry decomposition process for hexahedral mesh generation. Medial surface is used to recognize sweepable regions, and by highlighting the ideal cutting regions, the users can create cutting surfaces with the intuitive and user-friendly pen-based UI [7]. The models are then decomposed into sweepable sub-volumes. However the case that the medial surface degenerates is not considered, which may need special process. The common defect of medial surface based methods is that the generation of medial surface is very complex and the research of a robust medial surface generation algorithm is still undergoing.

Another kind of methods use the surface mesh to guide the volume decomposition. White et al. [8] use pseudo or virtual geometry to decompose complex volumes into mappable sub-volumes. The connectivity of the surface mesh is used to parameterize the surface nodes and this parameterization provides the information for decomposition of the volume into mappable sub-volumes by creating virtual surfaces inside the volume. The solid model is not decomposed physically as the virtual surfaces have no underlying geometry. The processible solid models of this method are limited to the sub-mappable ones. White et al. [9] automatically decompose multi-sweepable volumes into many-to-one sweepable volumes with the aid of structured side mesh, then many-to-one hexahedral sweeping approaches can be applied to generate hexahedral mesh. Blacker [10] proposes an automated hexahedral mesh generation approach named “cooper tool”. The tool recognizes applicable geometries and decomposes them into logically single axis swept sub-volumes (barrels). A well formed continuous mesh can be generated throughout the geometry with a set of processes. Miyoshi and Blacker [11] further extend cooper tool to multi-axis. Applicable geometries can be recognized and divided into a hierarchy of sub-volumes which can be meshed by existing single-axis sweep tools automatically. The above surface mesh based methods are all aimed at dealing with specific kinds of geometry, hence their generality is weak.

Feature recognition is also applied to volume decomposition. Liu and Gadh [12] present a recursive volume decomposition method which automatically decomposes complex shaped objects into simple sub-objects for automatic hexahedral mesh generation. Edge loops are used to determine basic simple objects. The sub-volumes after decomposition are classified into either convex or swept volumes, and there is a corresponding hexahedral meshing approach for each sub-volume. Lu et al. [13] present the work on shape recognition and volume decomposition to automatically decompose a CAD model into hex meshable volumes. Feature recognition technique is used to guide the decomposition in an intelligent way. Both methods have no discussion on how to ensure the mesh conformity between sub-volumes, especially when adjacent sub-volumes are meshed with different meshing algorithms, the conformity is difficult to guarantee.

CAD models are usually generated by a set of extrusion and rotation operations, each of which corresponds to a feature. So a CAD model can be viewed as the interaction result of all these features, and most of the geometry of the features are swept volumes. Sweeping is a hexahedral mesh generation algorithm that can be used to generate high quality hexahedral mesh for swept volumes. Of all the hexahedral mesh generation techniques, sweeping tends to be the workhorse algorithm, accounting for at least 50 percent of most meshing applications [14]. Therefore, some scholars have carried out a few volume decomposition works specially aiming at obtaining swept volumes.

Shih et al. [15] present an automated swept volume decomposition algorithm. Complex solid model is decomposed into simple swept volumes by recursive swept volume generation and removal. However, this approach requires large computational effort when the solid model is very complex, as it requires a large amount of Boolean operations. Besides, this method cannot deal with the case that the interactions between swept volumes are very complex. For example, when the generator face of the swept volume is destroyed by interaction, the corresponding swept volume cannot be generated.

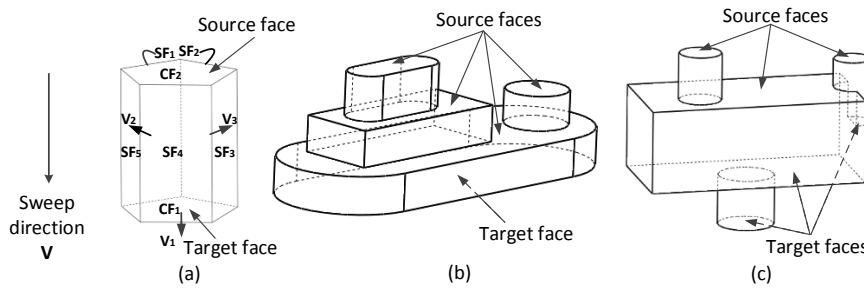


Fig. 1. Three types of swept volumes. (a) One-to-one swept volume. CF_1 and CF_2 are cap faces, $SF_1 \sim SF_5$ are side faces, V_1 , V_2 and V_3 are the normals of CF_1 , SF_1 and SF_2 respectively. (b) Many-to-one swept volume. (c) Many-to-many swept volume.

In this paper, according to the requirements of high quality hexahedral mesh generation, an automatic swept volume decomposition approach which can decompose a complex solid model into a set of simple volumes including all the swept volumes in the solid model is presented. Compared with existing approaches, our approach has the following characteristics:

- To identify all swept volumes including heavily interacting ones from the solid model, we extract all the potential local sweep directions (PLSDs) of the solid model first, then form a relevant face set (RFS) for each PLSD and determine all the swept volumes based on them step by step.
- To guarantee mesh quality, for interacting swept volumes, we construct a reasonable cutting face set (CFS) to separate them.
- To minimize the effort to guarantee mesh conformity in mesh generation stage, maximal single-axis swept volumes rather than arbitrary swept volumes are generated as the resultant swept volumes.

2. Basic Concepts and Approach Overview

2.1. Basic concepts

2.1.1. Swept volume

When a face is moved along a path, the locus of the points of the face defines a volume. This volume is called a swept volume [15]. The volume shown in Fig. 1(a) is an example of a simple swept volume. The starting and ending faces are called source face and target face respectively, and both of them are called cap faces. The linking face generated in the moving phase is called the side face. In the present work, we limit the source face to be planar and the moving path to be a straight line perpendicular to the source face. Thus the normal of cap face is the same with or opposite to the sweep direction, and side face is perpendicular to cap face.

Sweeping based hexahedral mesh generation algorithms can be classified into three categories: one-to-one, many-to-one and many-to-many. The corresponding swept volumes are with single source face and single target face, multiple source faces and single target face, multiple source faces and multiple target faces respectively, as shown in Fig. 1. The common characteristic of these three types of swept volumes is that they are all single-axis swept volumes.

In our approach, the resultant swept volumes after decomposition are all maximal single-axis swept volumes, that is to say, the geometry will not be further decomposed if it is already of one of the three types and adjacent swept volumes with the same sweep direction will be combined in the result.

Interacting swept volumes can either join each other at the boundary or have non-zero volume overlap between them, and we call the latter heavily interacting swept volumes. For example, the interacting swept volumes as shown in Fig. 2 are heavily interacting ones. It can be easily noticed that the original swept volumes are hard to be restored from such interacting volumes, due to the fact that the sweep characteristics are heavily destroyed.

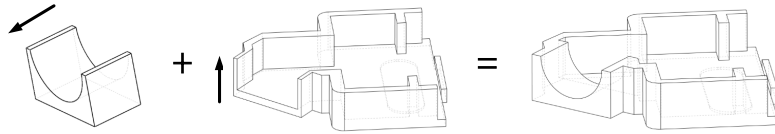


Fig. 2. Interacting swept volumes.

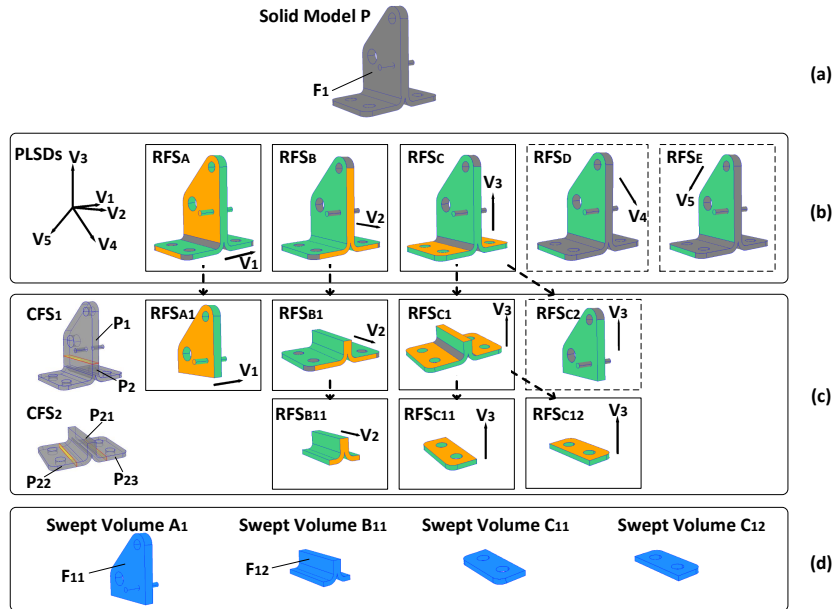


Fig. 3. The process of this approach. (a) Original solid model P. (b) Extracted potential local sweep directions (PLSDs) and generated relevant face set (RFS) for each PLSD. (c) Cutting face sets (CFSs) constructed to separate interacting RFSs, and reformed RFSs. (d) Resultant swept volumes. (For the RFSs, cap faces are colored in orange and side faces are colored in green. Component faces of CFSs are colored in yellow, and their boundary edges are highlighted by red to show more clearly.)

2.1.2. Potential local sweep direction (PLSD)

Potential local sweep directions (PLSDs) are used to represent the sweep directions of the potential swept volumes that may result from the swept volume decomposition process. A PLSD is represented by a vector, as the sweep path is confined to straight line in the current approach. For example, $V_1 \sim V_5$ represent five different PLSDs in Fig. 3.

2.1.3. Relevant face set (RFS) of PLSD

Given a PLSD, the faces of the solid model that could be the side face or cap face for this sweep direction can be all identified. These faces are called the relevant face set (RFS) of this PLSD. RFS may not form a valid swept volume directly, but swept volumes can be generated by adding missing faces and deleting redundant face parts from it. There are some examples of RFSs in Fig. 3.

2.1.4. Cutting face set (CFS)

Cutting face set (CFS) is a set of faces that can split the solid model into several sub-volumes. For a CFS, its global structure can be manifold or non-manifold, its component faces can be connected or disconnected, and its outer boundary edges must lie on the surface of the solid model. For example, in Fig. 3, CFS₁ and CFS₂ are CFSs. CFS₁ is composed of only one face, and CFS₂ is composed of two disconnected faces.

Different RFSs may share faces. For example, the interaction of RFS_A and RFS_B in Fig. 3 is shown in Fig. 4(a). The red arrow and blue arrow represent the sweep directions of RFS_A and RFS_B respectively. Classifying the faces

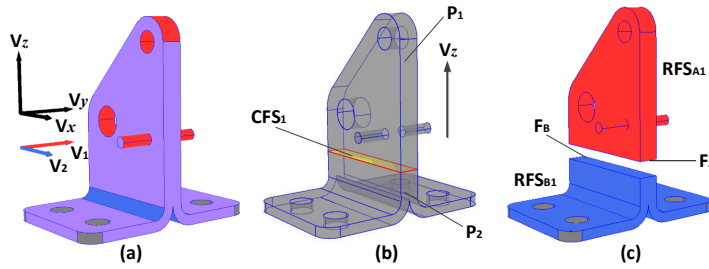


Fig. 4. (a) Interaction between RFS_A and RFS_B in Fig. 3. The purple faces are shared faces. V_x, V_y and V_z are three possible cutting face normals. (b) The solid model P is decomposed into P₁ and P₂ by CFS₁. (The component face of CFS₁ is colored in yellow, and its boundary edges are highlighted by red to show more clearly.) (c) RFS_A and RFS_B are reformed to RFS_{A1} and RFS_{B1} respectively.

of solid model P by their relationship with RFS_A and RFS_B, each face f must fall into one of the following four categories:

- **Category (1):** $f \in \text{RFS}_A$ and $f \notin \text{RFS}_B$;
- **Category (2):** $f \notin \text{RFS}_A$ and $f \in \text{RFS}_B$;
- **Category (3):** $f \in \text{RFS}_A$ and $f \in \text{RFS}_B$;
- **Category (4):** $f \notin \text{RFS}_A$ and $f \notin \text{RFS}_B$.

In Fig. 4(a), the red faces belong to Category (1), the blue faces belong to Category (2), the purple faces belong to Category (3), and the gray faces belong to Category (4).

A CFS is constructed to separate two RFSs who share faces by splitting the solid model. In Fig. 4(b), CFS₁ is constructed and decomposes P into P₁ and P₂. To separate RFS_A and RFS_B, faces in Category (1) must all lie on P₁, faces in Category (2) must all lie on P₂, faces in Category (3) and (4) can lie on either P₁ or P₂. Then RFS_A can be reformed to RFS_{A1} on P₁, and RFS_B can be reformed to RFS_{B1} on P₂. RFS_{A1} and RFS_{B1} do not share faces any more. Besides, CFS₁ corresponds to a new set of faces CFS_A on P₁, and corresponds to a new set of faces CFS_B on P₂. We require $\text{CFS}_A \subset \text{RFS}_{A1}$ (i.e. all the faces of CFS_A belong to RFS_{A1}) and $\text{CFS}_B \subset \text{RFS}_{B1}$. That is to say, all the component faces of CFS₁ should be able to be the side face or cap face of both RFS_A and RFS_B. Finally, faces in Category (1) and Category (3) and CFS_A form RFS_{A1} on P₁, and faces in Category (2) and Category (3) and CFS_B form RFS_{B1} on P₂.

In Fig. 4(b), CFS₁ is composed of only one face and introduces two new faces F_A and F_B on P₁ and P₂ respectively as shown in Fig. 4(c). F_A is a side face of RFS_{A1}, and F_B is a side face of RFS_{B1}. It can be noticed that after the construction of CFS₁, RFS_A and RFS_B are reformed to be more complete to form swept volumes with redundant faces discarded and new cutting faces included.

It should be noted that after CFS construction, the solid model can be decomposed into more than two sub-volumes and in this case the RFSs are reformed in the similar way.

2.2. Approach overview

In order to effectively support automatic and high quality hexahedral meshing of complex solid models, we put forward an approach to automated swept volume decomposition of a solid model based on sweep directions extraction. The input of the approach is a B-rep solid model and the output is a set of simple volumes including all the swept volumes in the solid model.

The automated swept volume decomposition faces two critical issues. One is how to effectively recognize those heavily interacting swept volumes that are usually hard to recognize for their sweep characteristics may be destroyed to a great extent by interaction. For example, in Fig. 2, two swept volumes heavily interact each other and their surfaces are destroyed. It is hard to recognize the original swept volumes from the resultant volume. Another issue is how to split the interacting swept volumes so that every separated swept volume can be well meshed, i.e. their hexahedral mesh quality can be guaranteed. In our approach, we solve the first issue by extracting all the potential

local sweep directions (PLSDs) and incrementally determining all the swept volumes based on them. This works because for any interacting swept volume, its sweep direction can always be determined according to the remaining faces of the swept volume in the solid model. Regarding the second issue, our approach tackles it by constructing a reasonable cutting face set (CFS) to separate interacting swept volumes with mesh quality as optimization objective.

As illustrated by Fig. 3, our approach consists of following three steps:

1. **Extraction of all potential local sweep directions (PLSDs) and generation of relevant face sets (RFSs).**
Extract all the PLSDs using heuristic rules and generate the RFS of each PLSD.
2. **Construction of reasonable cutting face set (CFS) for interacting swept volumes.**
Construct reasonable CFS and use it to separate the interacting swept volumes.
3. **Generation of maximal single-axis swept volumes.**
Generate maximal single-axis swept volumes by each RFS.

3. Extraction of all Potential Local Sweep Directions and Generation of Relevant Face Sets

The extraction of all potential local sweep directions (PLSDs) is the basis step of this approach, and PLSDs will be used to guide the swept volume decomposition process. The main problem of PLSDs extraction is that swept volumes may be not complete due to volume interaction, and only partial faces of interacting swept volumes remain on the solid model. Fortunately, the sweep direction of an interacting swept volume can be determined by its remaining faces by making use of the relationship between the sweep direction and the faces of swept volume. In order to extract all the PLSDs, we analyse all the faces of the solid model according to the three heuristic rules listed in Table 1. As the sweep paths of swept volumes are confined to straight lines in the current approach, the faces we deal with are restricted to planes and ruled surfaces.

Table 1. Heuristic rules to extract PLSDs.

Rules	Description	Illustration
Rule-1	If a face is planar, then its normal is a PLSD.	Sweep direction is parallel or antiparallel to the normal of cap face. For example, the cap face normal V_1 is parallel to the sweep direction V , as shown in Fig. 1(a).
Rule-2	If the normals of two planar faces are not parallel or antiparallel to each other, then their cross product is a PLSD.	Side face normal is perpendicular to sweep direction. Therefore the sweep direction can be determined by two planar side faces. As shown in Fig. 1(a), the normal V_2 of side face SF_1 and the normal V_3 of side face SF_2 are both perpendicular to sweep direction V , hence the sweep direction V can be calculated by $V_2 \times V_3$ or $-V_2 \times V_3$. (V_2 and V_3 are not parallel or antiparallel)
Rule-3	If the generatrices of a ruled surface are parallel to each other, then their direction is a PLSD.	Curved side face of swept volume is generated from a curve of source face, and the sweep path is straight line, hence curved side face must be ruled surface whose generatrices are parallel to each other. The direction of the generatrices is parallel to the sweep direction, thus the sweep direction can be obtained.

According to the above rules, the sweep direction of a swept volume can be obtained from only partial faces of it as long as at least one of the following three conditions is satisfied: first, at least one cap face remains; second, at least two planar side faces remain; third, at least one curved side face remains. In fact, these conditions include all the situations but two: first, only one planar side face of the swept volume remains; second, no face of the swept volume remains. In both cases, almost all the faces of the swept volume disappear in the solid model, and this usually indicates that the swept volume is contained by other volumes during interaction, and there is no need to extract this swept volume. Therefore the above method can extract PLSDs to the most extent.

A bunch of PLSDs can be obtained by analysing all the faces of the solid model. However, some of them may be parallel to each other, since for one swept volume, its sweep direction can be determined by both two side faces and one cap face. Besides, antiparallel vectors are identical when used to represent the sweep directions of potential swept volumes (the sweep direction of a swept volume can be represented by an antiparallel vector by switching the roles of source and target faces without changing the swept volume itself). Therefore parallel and antiparallel PLSDs are clustered and only one of them is reserved. The implementation details of PLSDs extraction method used in this paper are shown in Algorithm 1.

Algorithm 1 PLSDs extraction algorithm

Input: s : solid model.
Output: P : the set that stores PLSDs.

```

1: Initialize a new set  $A$ ;
2: Initialize a new set  $B$ ;
3: for all planar face  $p$  of  $s$  do
4:   Add the normal of  $p$  to  $A$ ;
5:   Add the normal of  $p$  to  $B$ ;
6: end for
7: for  $i = 1$  to  $size(B)$  do
8:   for  $j = i + 1$  to  $size(B)$  do
9:     Add  $B_i \times B_j$  to  $A$ ;
10:   end for
11: end for
12: for all curved face  $c$  of  $s$  do
13:   if  $c$  is a ruled surface whose generatrices are parallel to each other then
14:     Add the direction of the generatrices to  $A$ ;
15:   end if
16: end for
17: Cluster the vectors in  $A$  that are parallel or antiparallel to each other;
18: for all cluster do
19:   Add one of the vectors to  $P$ ;
20: end for

```

It should be pointed out that one PLSD does not necessarily correspond to one swept volume in the final decomposition result. First, there may be more than one swept volumes corresponding to a PLSD. This is true because different swept volumes may have the same sweep direction. Second, there may be no swept volume corresponding to a PLSD. There are two reasons for this: One is that the potential swept volume for this PLSD is contained by other swept volumes, thus only the larger swept volumes appear in the final decomposition result; the other is that the potential swept volume cannot be fully extracted when dealing with the interaction between swept volumes.

After extracting all the PLSDs, the relationship between the faces and the sweep direction of swept volume can be directly translated into an algorithm to form a relevant face set (RFS) for each PLSD. For the solid model in Fig. 3, five different PLSDs are extracted, and RFS_A , RFS_B , RFS_C , RFS_D and RFS_E are corresponding RFSs.

It can be noted that one face of the solid model may belong to more than one RFSs, and this causes the interaction between RFSs. There are two reasons for this phenomenon: First, the faces of two swept volumes are merged into one face after interaction, hence this merged face belongs to two RFSs. This face should be split into two RFSs by CFS construction and finally each part of this face belongs to one swept volume. As shown in Fig. 3, F_1 belongs to both RFS_A and RFS_B . F_1 is split into F_{11} and F_{12} after a set of CFSs construction, and finally F_{11} belongs to Swept Volume A_1 , F_{12} belongs to Swept Volume B_{11} . Second, since RFS is formed just according to the sweep direction, many faces are still included by the RFS because they meet the requirement of the sweep direction although they do not contribute to the generation of the final resultant swept volumes. That is to say, there are redundant faces inside RFSs. These redundant faces will be deleted from the RFS and assigned to the most suitable one by CFSs construction. As shown in Fig. 3, F_1 also belongs to RFS_C initially, but it is deleted from it after a set of CFSs construction. Finally F_1 does not contribute to the generation of Swept Volume C_{11} or C_{12} . The CFS construction algorithm will be discussed in the next section.

4. Construction of Reasonable Cutting Face Set for Interacting Swept Volumes

Two different RFSs may share faces, and in order to generate non-intersecting swept volumes with them, one cutting face set (CFS) is constructed to separate every two interacting RFSs. There are two key problems for constructing a CFS. First, there may be more than one CFSs that can be used to separate two RFSs. For example, in Fig. 4(b), different CFSs can be constructed by moving CFS_1 a small distance along V_z or reverse to V_z , and all these CFSs can be used to separate RFS_A and RFS_B . Second, the interaction between RFSs can be so complex that the structure of the CFS used to separate them has to be very complex. To solve the two problems, an algorithm to construct a reasonable CFS which can guarantee that high quality hexahedral mesh can be generated after the decomposition is proposed.

In fact, to separate RFS_A and RFS_B by constructing a CFS, the intrinsic thing to do is to divide Category (1) faces and Category (2) faces into different sub-volumes. To construct the reasonable CFS, we employ a greedy strategy to construct one optimal cutting face to separate some faces of Category (1) and some faces of Category (2) at a time, and repeat this process iteratively until each sub-volume does not contain Category (1) faces and Category (2) faces at the same time. Then combine the sub-volumes that contain Category (1) faces and Category (2) faces respectively. The joint faces between the two types of sub-volumes compose the CFS, and separated RFSs can be reformed on the resultant sub-volumes. However, if no eligible cutting face can be constructed during the iterative decomposition process, then the CFS construction fails. There could be two reasons for this: one is that a reasonable CFS does not exist, the other is that the greedy strategy fails to construct a reasonable CFS. In our current experiments, the failure of this greedy strategy rarely happens.

The process of CFS construction is summarized in Algorithm 2. To determine whether sub-volume Q_i needs decomposition, it needs only to determine whether it contains faces of Category (1) and (2) at the same time. If Q_i contains both Category (1) faces and Category (2) faces, then it needs decomposition, otherwise not. Some details of this algorithm will be further explained in Sect.4.1 to Sect.4.3.

Algorithm 2 CFS construction algorithm

```

1: Put the volume  $p$  that the two RFSs lie on in a queue  $Q$ ;
2: Determine all possible cutting face normals;
3: while  $Q$  is not empty do
4:   Take out one volume  $Q_i$  from  $Q$ ;
5:   if  $Q_i$  needs decomposition then
6:     for all possible cutting face normal  $V$  do
7:       Determine a set of candidate cutting locations and construct corresponding cutting faces with them on  $Q_i$ ;
8:     end for
9:     Select the optimal cutting face  $c$  from all the cutting faces according to the evaluation function;
10:    if  $c$  meets requirements then
11:      Decompose  $Q_i$  with  $c$  and put the resultant sub-volumes in queue  $Q$ ;
12:    else
13:      CFS construction fails;
14:      return
15:    end if
16:  else
17:    Put  $Q_i$  in queue  $R$ ;
18:  end if
19: end while
20: for all volumes in queue  $R$  do
21:   Combine all adjacent volumes that contain Category (1) faces and no Category (2) faces;
22:   Combine all adjacent volumes that contain Category (2) faces and no Category (1) faces;
23: end for
24: The joint faces between the resultant volumes in  $R$  compose the CFS.

```

The solid model is decomposed in the meantime of CFS construction, therefore the process to separate two RFSs is in fact eliminating the interaction between potential swept volumes by decomposing the solid model. Fig. 5 gives an example of CFS construction. The interaction of RFS_M and RFS_N is shown in Fig. 5(c). CFS_{MN} which is composed of three connected faces as shown in Fig. 5(f) is constructed. RFS_M and RFS_N are reformed to RFS_{M1} in Fig. 5(d) and RFS_{N1} in Fig. 5(e) respectively, and do not share faces any more.

4.1. Determination of cutting face normals

We require every component face of a CFS to be able to be the side face or cap face of both RFSs it deals with, in other words, a cutting face F should meet the requirements of the sweep directions of both RFSs. Cutting face normals are determined by this restriction. Curved face can only be side face in our approach, and the sweep direction of corresponding swept volume is fixed. Therefore curved face cannot meet the sweep direction requirements of both RFSs at the same time. Only planar face can be cutting face.

Assume the sweep directions of RFS_A and RFS_B are V_a and V_b respectively, then the normal of cutting face F should meet the following two requirements: (1) perpendicular or parallel to V_a , (2) perpendicular or parallel to V_b .

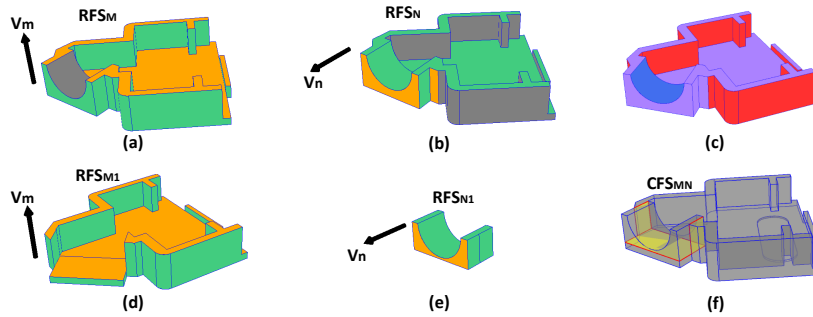


Fig. 5. (a) RFS_M and its sweep direction V_m . (b) RFS_N and its sweep direction V_n . (c) Interaction between RFS_M and RFS_N . The red faces belong to Category (1), the blue faces belong to Category (2), the purple faces belong to Category (3), and the gray faces belong to Category (4). (d) RFS_M is reformed to RFS_{M1} . (e) RFS_N is reformed to RFS_{N1} . (f) CFS_{MN} . (For the RFSs, faces in orange are cap faces, and faces in green are side faces.)

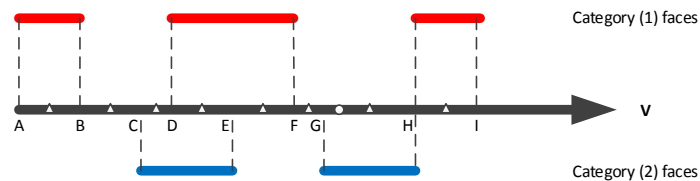


Fig. 6. Determination of cutting locations for cutting face normal V .

So, if V_a and V_b are perpendicular to each other, then the possible normals of F are (1) V_a , (2) V_b , (3) $V_a \times V_b$; else the only possible normal of F is $V_a \times V_b$. For the example in Fig. 4(a), V_x , V_y and V_z can all be the cutting face normals.

4.2. Determination of cutting face location

Given a cutting face normal, different cutting faces can be constructed at different locations, and they have different degrees of ability to separate faces of Category (1) and (2). To cover all the separation cases, a set of cutting locations is determined for each possible cutting face normal.

Project faces of Category (1) and (2) to the cutting face normal V , as shown in Fig. 6, each line segment represents the projection result of one face. The red ones are the projection results of Category (1) faces, and the blue ones are the projection results of Category (2) faces. These line segments form a list of intervals, denoted by AB, BC, CD and so on. Then two types of cutting locations are defined for a cutting face normal V :

- **Type-1:** *Midpoint of each interval.* The cutting faces constructed at any internal point of one interval are identical in the aspect of separating the faces of Category (1) and (2). For example, in Fig. 6 the cutting faces constructed at any internal point of BC are identical. For all of them, only one Category (1) face is on the left side, and residual Category (1) faces and all Category (2) faces are on the right side. The midpoints of all the intervals cover all the separation cases for faces of Category (1) and (2).
- **Type-2:** *Projection points of the solid model's faces to V , whose normals are the same with or opposite to V .* Using these projection points as cutting locations means utilizing the faces of the original solid model as cutting faces, and this helps to reduce the introduction of new faces.

For example, in Fig. 6 the points denoted by triangles represent Type-1 cutting locations, and the point denoted by circle represents a Type-2 cutting location. They are all candidate cutting locations for cutting face normal V , and one cutting face can be constructed at each of them. A set of cutting faces can be constructed at all the candidate cutting

locations for all the possible cutting face normals. Among all these cutting faces, one optimal cutting face evaluated by the evaluation function presented in Section 4.3 will be selected as the real cutting face.

4.3. Evaluation function for cutting face

The cutting faces are evaluated with five criteria, denoted by C , A , D , S and I . These criteria evaluate the cutting faces in different aspects. Some criteria evaluate the ability to separate the two RFSs, and some criteria evaluate the ability to generate high quality hexahedral mesh after the decomposition. The ideal optimal cutting face should outperform others when evaluated with any one of the five criteria, however, usually this ideal cutting face does not exist. It is a multi-objective optimization problem to select the optimal cutting face. Each evaluation criterion is assigned a weight and the evaluation function defined below is used to evaluate the cutting faces:

$$E = w_C \cdot C + w_A \cdot A + w_D \cdot D + w_S \cdot S + w_I \cdot I \quad (1)$$

in which, w_C, w_A, w_D, w_S and w_I are the weights for each criterion and $w_C + w_A + w_D + w_S + w_I = 1$. The larger E is, the better the cutting face is. The detailed description of each criterion is listed below:

- The ability to separate Category (1) and Category (2) faces

$$C = \frac{\sum_{i=1}^n |P_{i1} - P_{i2}|}{\sum_{i=1}^n |P_{i1} + P_{i2}|} \quad (2)$$

in which $n(n \geq 2)$ is the number of sub-volumes that result from the construction of this cutting face, P_{i1} is the number of Category (1) faces on the i th sub-volume, P_{i2} is the number of Category (2) faces on the i th sub-volume. Since the intrinsic function of CFS is to separate Category (1) and Category (2) faces, the optimal cutting face should separate as much Category (1) and Category (2) faces as possible. The larger C is, the stronger the ability of this cutting face to separate Category (1) and Category (2) faces is, and the weaker on the contrary. When $C=1$, this cutting face can separate all Category (1) faces from all Category (2) faces.

- The introduced worst dihedral angle between faces

$$A = \begin{cases} \sin \alpha & \alpha < \pi/2 \\ 1 & \alpha \geq \pi/2 \end{cases} \quad (3)$$

in which α is the minimal dihedral angle introduced by cutting face construction. Extreme dihedral angle is bad for the generation of high quality mesh and should be avoided being introduced by cutting face construction. The smaller A is, the smaller the minimal dihedral angle introduced is, and it is harder to generate high quality mesh. When $A=1$, the newly introduced dihedral angles bring no trouble.

- The introduced worst distance between faces

$$D = \begin{cases} d/m & d < m \\ 1 & d \geq m \end{cases} \quad (4)$$

in which d is the newly introduced minimal distance between faces and m is the final element size assigned by the user. Extreme distance between faces is bad for the generation of high quality mesh and should be avoided being introduced by cutting face construction. The smaller D is, the smaller the newly introduced minimal distance between faces is, and it is harder to generate high quality mesh. When $D=1$, the newly introduced distances between faces bring no trouble.

- Whether it is coplanar with a face of the solid model. If the cutting face is coplanar with a face of the solid model, then $S=1$, and otherwise $S=0$. Using the face of the original solid model as cutting face helps to reduce the introduction of new faces, thus reduces the constraints brought in by volume decomposition for mesh generation.
- Whether it intersects with Category (4) faces. If the cutting face intersects with Category (4) faces, then $I=0$, and otherwise $I=1$. Cutting face should avoid intersecting the faces that belong to neither RFS it deals with, so as to reduce the impact on other RFSs.

The above criteria are listed in the order of descending importance. In our implementation, the values of w_C , w_A , w_D , w_S and w_I are set as 0.3, 0.3, 0.3, 0.05 and 0.05 respectively. All the cutting faces constructed above are evaluated with this evaluation function and the optimal one with the largest evaluation value is selected as the real cutting face. Besides, according to experiments, if the largest evaluation value is less than 0.6, this means the optimal cutting face does not meet the requirements and no cutting face will be constructed, thus CFS construction fails.

5. Generation of Maximal Single-Axis Swept Volumes

Before RFSs can be used to generate non-intersecting swept volumes, a set of CFSs should be constructed to separate all the interacting RFSs. We construct one CFS at a time, and decompose the solid model bi-recursively. To determine which CFS is constructed each time, the concept of interaction degree between RFSs is defined. Starting from the two RFSs with the highest interaction degree, we try to construct a CFS until a reasonable one is successfully constructed. The interaction degree between RFS_A and RFS_B is defined as:

$$I_{AB} = \frac{I_F}{A_F + B_F - I_F} \quad (5)$$

in which I_F is the number of shared faces, A_F is the number of faces of RFS_A, B_F is the number of faces of RFS_B. The larger I_{AB} is, the higher the interaction degree between RFS_A and RFS_B is.

Each time a CFS is constructed, the solid model is further decomposed, the corresponding two RFSs are reformed, and all the other RFSs are also split accordingly. Besides, whenever all the faces of one RFS are already contained by other RFSs, i.e. the RFS has no face that only belongs to itself, it is discarded. This helps to filter out the potential swept volumes that may be contained by other swept volumes.

The solid model is decomposed into a collection of sub-volumes after the construction of a set of CFSs, and finally there are one or more RFSs on each sub-volume. These RFSs are used to generate swept volumes. In fact, if a sub-volume contains only one RFS, and the faces of this RFS can wrap the sub-volume, then the sub-volume itself is a swept volume. In other cases, by extending all the faces of a RFS, the corresponding sub-volume is further divided into a set of smaller parts. For each part, if all its faces can be classified as cap face or side face, it is sweepable. All the sweepable parts compose the swept volumes of this RFS. If there are more than one RFSs on a sub-volume, then the swept volumes generated from different RFSs may intersect each other. In this case, the larger swept volumes are kept and the smaller ones are discarded. Finally, adjacent swept volumes with the same sweep direction are combined to generate maximal single-axis swept volumes. This helps to decrease the number of resultant volumes, thus reduce the effort to guarantee mesh conformity during mesh generation stage.

For example, in Fig. 3, initially five RFSs are formed, and RFS_D and RFS_E are discarded because all their faces are contained by other RFSs. First, CFS₁ is constructed to separate RFS_A and RFS_B, and the solid model is decomposed into P₁ and P₂. RFS_A and RFS_B are reformed to RFS_{A1} and RFS_{B1} respectively, and RFS_C is split into RFS_{C1} and RFS_{C2}. Since RFS_{C2} is contained by RFS_{A1} completely, it is discarded. Second, CFS₂ is constructed to separate RFS_{B1} and RFS_{C1}, and P₂ is decomposed into P₂₁, P₂₂ and P₂₃. RFS_{B1} is reformed to RFS_{B11}, and RFS_{C1} is reformed into RFS_{C11} and RFS_{C12}. Now four RFSs (RFS_{A1} on P₁, RFS_{B11} on P₂₁, RFS_{C11} on P₂₂ and RFS_{C12} on P₂₃) remain and they do not share faces, hence no more CFSs need to be constructed. These RFSs all wrap their corresponding sub-volumes, therefore these sub-volumes are swept volumes, and A₁, B₁₁, C₁₁ and C₁₂ are the swept volumes we get finally.

6. Experimental Results

The proposed approach is implemented and a bunch of solid models are tested to verify its efficiency. Some of them are shown in Fig. 7. The original solid models, the decomposition results, the time cost, the number of swept volumes and the number of residual volumes are all presented. The swept volumes are shown in different colors, and the residual volumes are colored in red.

The solid models in Example 3 and Example 4 are decomposed by extending the faces of the original solid models. No new geometry is introduced by the decomposition process, thus less constraints are brought in for the mesh


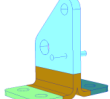

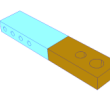
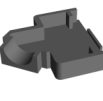
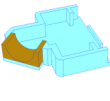
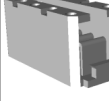
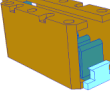

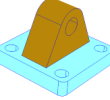
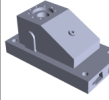
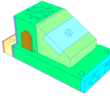
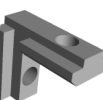
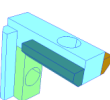

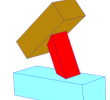
No.	Solid Model	Decomposition Result	Time (in sec)	Swp Vols Num	Residual Vols Num	No.	Solid Model	Decomposition Result	Time (in sec)	Swp Vols Num	Residual Vols Num
1			0.14	4	0	5			0.04	2	0
2			0.24	2	0	6			0.91	3	0
3			0.05	2	0	7			0.85	10	0
4			0.14	4	0	8			0.08	2	1

Fig. 7. Some examples of swept volume decomposition.

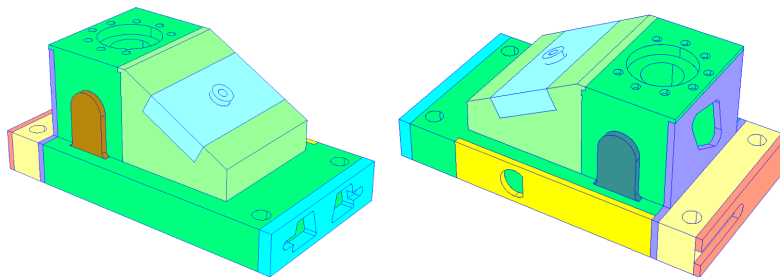


Fig. 8. Magnified front and back views of the decomposition result of the solid model in Example 7.

generation stage. The solid model in Example 5 cannot be well decomposed without introducing a new face. All the generator faces of swept volumes are merged with other faces by swept volume interaction, and with all the generator faces destroyed, previous approach based on swept volume generation [15] fails to decompose this solid model. In our approach, an optimal cutting face is constructed and the solid model is decomposed into two swept volumes. In Example 6, the solid model is composed of 147 faces and it is a bit complex, however, only three swept volumes are returned from the swept volume decomposition process, since the objective swept volumes of our approach are maximal single-axis swept volumes. The number of resultant volumes is greatly reduced, thus less effort will be needed to guarantee mesh conformity in the mesh generation stage. The solid model in Example 7 is rather complex. The interaction between swept volumes destroys a great deal of faces, therefore previous approach based on swept volume generation [15] cannot be applied. Besides, the generation of medial surface is very complex for this solid model, hence previous approach based on medial surface [5,6] is also not appropriate. Our approach decomposes it into ten swept volumes automatically in less than one second. The magnified front and back views of the decomposition result are shown in Fig. 8. The decomposition result of the solid model in Example 8 consists of two swept volumes and one residual volume. The two swept volumes are cubes. The residual volume is a prism whose two cap faces incline a bit. Although it can be meshed with the sweeping algorithm, however, it is not recognized as a swept volume, since it is not a strict swept volume previously defined. In fact, the sweep direction of the prism has been extracted by our approach. In the future, this kind of volumes will be easily recognized as swept volumes by enriching the categories of swept volumes.

7. Conclusion and Future Work

In this paper, we have proposed a novel approach to automated swept volume decomposition of solid models for high quality hexahedral meshing. Compared with existing methods, the proposed approach offers the following advantages:

- Be able to recognize heavily interacting swept volumes from a solid model. This is achieved by extracting all the potential local sweep directions first and then determining all the swept volumes based on them step by step.
- For the interacting swept volumes that can be decomposed in different ways, a reasonable cutting face set is constructed to decompose them. In this way, the quality of the hexahedral mesh generated from each decomposed swept volume can be guaranteed.

The swept volumes are currently limited to the ones whose sweep paths are straight lines and cap faces are perpendicular to the sweep directions. In the future, we will extend the current approach to deal with the pseudo swept volumes [16]. Besides, as our current greedy strategy to construct a reasonable CFS may fail, we will provide a better solution which can find a reasonable CFS as long as it exists. In addition, hexahedral meshing based on the swept volume decomposition results generated by our approach will be conducted.

Acknowledgements

The authors are very grateful to the financial support from 863 High Tech. Plan of China (2013AA041301).

References

- [1] T. Blacker, Automated conformal hexahedral meshing constraints, challenges and opportunities, *Engineering with Computers*, 17.3 (2001) 201–210.
- [2] H. Blum, A transformation for extracting new descriptors of shape, *Models for the perception of speech and visual form*, 19.5 (1967) 362–380.
- [3] M.A. Price, C.G. Armstrong, M.A. Sabin, Hexahedral mesh generation by medial surface subdivision: Part I. Solids with convex edges, *International Journal for Numerical Methods in Engineering*, 38.19 (1995) 3335–3359.
- [4] M.A. Price, C.G. Armstrong, Hexahedral mesh generation by medial surface subdivision: Part II. Solids with flat and concave edges, *International Journal for Numerical Methods in Engineering*, 40.1 (1997) 111–136.
- [5] J.H.C. Lu, I. Song, W.R. Quadros, K. Shimada, Volumetric Decomposition via Medial Object and Pen-Based User Interface for Hexahedral Mesh Generation, *Proceedings of the 20th International Meshing Roundtable*, Springer Berlin Heidelberg, 2012, pp. 179–196.
- [6] J.H.C. Lu, I. Song, W.R. Quadros, K. Shimada, Geometric reasoning in sketch-based volumetric decomposition framework for hexahedral meshing, *Proceedings of the 21st International Meshing Roundtable*, Springer Berlin Heidelberg, 2013, pp. 297–314.
- [7] J.H.C. Lu, I. Song, W.R. Quadros, K. Shimada, Pen-based user interface for geometric decomposition for hexahedral mesh generation, *Proceedings of the 19th International Meshing Roundtable*, Springer Berlin Heidelberg, 2010, pp. 263–278.
- [8] D.R. White, L. Mingwu, S.E. Benzley, G.D. Sjaardema, Automated hexahedral mesh generation by virtual decomposition, *Proceedings of the 4th International Meshing Roundtable*, Sandia National Laboratories, Albuquerque, USA, 1995, pp. 165–176.
- [9] D.R. White, S. Saigal, S.J. Owen, CCSweep: automatic decomposition of multi-sweep volumes, *Engineering with computers*, 20.3 (2004) 222–236.
- [10] T. Blacker, The cooper tool, *Proceedings of the 5th International Meshing Roundtable*, SAND 95-2130, Sandia National Laboratories, 1996.
- [11] K. Miyoshi, T.D. Blacker, Hexahedral Mesh Generation Using Multi-Axis Cooper Algorithm, *Proceedings of the 9th International Meshing Roundtable*, 2000, pp. 89–97.
- [12] S.S. Liu, R. Gadh, Automatic hexahedral mesh generation by recursive convex and swept volume decomposition, *Proceedings of the 6th International Meshing Roundtable*, 1997, pp. 217–231.
- [13] Y. Lu, R. Gadh, T.J. Tautges, Volume Decomposition and Feature Recognition for Hexahedral Mesh Generation, *Proceedings of the 8th International Meshing Roundtable*, 1999, pp. 269–280.
- [14] D.R. White, T.J. Tautges, Automatic scheme selection for toolkit hex meshing, *International Journal for Numerical Methods in Engineering*, 49.1 (2000) 127–144.
- [15] B.Y. Shih, H. Sakurai, Automated hexahedral mesh generation by swept volume decomposition and recombination, *Proceedings of the 5th International Meshing Roundtable*, 96 (1996) 273–280.
- [16] S.S. Liu, J. Uicker Jr, R. Gadh, A dual geometry–topology constraint approach for determination of pseudo-swept shapes as applied to hexahedral mesh generation, *Computer-Aided Design*, 31.6 (1999) 413–426.